

REMARKS

This is intended as a full and complete response to the Final Office Action dated August 9, 2006, having a shortened statutory period for response set to expire on November 9, 2006. Applicants submit this response to place the application in condition for allowance or in better form for appeal. Please reconsider the claims pending in the application for reasons discussed below.

Claims 1-3, 5-7, 9-12, 14, 15, 17-23, 25, 26, 28 and 29 are pending in the application.

Double Patenting

Claims 1-3, 5-7, 9-12, 14-15, 17-23, 25-26 and 28-29 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-34 of co-pending Application No. 10/037,595. Applicants acknowledge the double patenting rejection originally made in the Office Action mailed April 6, 2006, and respectfully request that the rejection be held in abeyance because (i) no claim in the present application is currently allowable and (ii) the application on which the rejection is made (No. 10/037,595) has not issued. Because it is possible that no claims will issue, or that the claims of the present application will be amended in such a way to overcome the Examiner's concerns regarding double patenting, Applicants defer responding until the present rejection ripens into an actual double patenting rejection.

Claim Rejections - 35 U.S.C. § 103

Claims 1-7, 9-11, 14-15, 17-18, 20-22, 25, 26, and 28-29 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Nair* (U.S. 2003/0217184) in view of *Beighe* (U.S. 6,055,576) in view of *Putcha* (U.S. 6,822,966).

Applicants respectfully traverse this rejection.

The Examiner bears the initial burden of establishing a *prima facie* case of obviousness. See MPEP § 2142. To establish a *prima facie* case of obviousness three basic criteria must be met. First, there must be some suggestion or motivation, either in

the references themselves or in the knowledge generally available to one ordinary skill in the art, to modify the reference or to combine the reference teachings. Second, there must be a reasonable expectation of success. Third, the prior art reference (or references when combined) must teach or suggest all the claim limitations. See MPEP § 2143. The present rejection fails to establish at least the third criteria.

Applicants submit that *Nair*, *Beighe*, and *Putcha* do not teach or suggest all of the limitations of the present claims. For example, these references do not teach a method of processing messages that includes “receiving, at a socket configured for a server application executing on a computer, data from a remote source via a network connection prior to allocating a buffer to contain the data.” Claims 9 and 20 recite a similar limitation.

Nair is directed to a process for “communications protocol software modules” to manage a data frame being passed across successive levels of a protocol stack using a pointer. *Nair*, Abstract.

The “protocol software modules” discussed in *Nair* are include the TCP (and lower) layers of a TCP/IP stack and further, *Nair* expressly distinguishes the operations of these “protocol software modules” from those of a “higher level application.” The following passages from *Nair* illustrate this point:

a hardware interface, typically implemented in a chipset, provides a physical connection to the network. A driver, such as ATM driver 105, transmits and receives information, generally in the form of a well defined stream of binary digits, respectively to and from the hardware interface 103. The driver provides a mechanism to transmit and receive the stream of binary digits as a block of data, whether defined as a fixed length cell, as in the case of an ATM stream of data, or a variable length frame of data, as in the case of an Ethernet-based frame of data transmitted over a local area network. An Ethernet/IEEE 802.3 hardware interface, not shown, provides a physical connection to local area network 101 and essentially operates in the same manner to generally perform the same functions as ATM hardware interface 103.

The ATM driver services higher layer protocol software modules in protocol stack implemented in the machine, such as PPP over ATM

adaptation layer 5 (PPP over AAL5) software module 107 and Point to Point Protocol (PPP) software module 109. These modules, in turn, service, for example, a higher layer protocol software module such as IP software module 110. Likewise, the Ethernet driver services the IP software module 110. Finally, IP software module 110 services TCP software module 112.

Nair, ¶ 18-19. These passages describe elements of *Nair*, Figure 1, which illustrate a physical layer (H/W I/F 103), a data link layer (Ethernet 108 or ATM driver 105 and PPP over AALS 109), a network layer (IP 110), and a transport layer (TCP 11) of a network protocol stack. Figure 1 does not even illustrate the “higher level application;” instead, this figure simply includes an arrow leading from the TCP module 112.

Thus, Applicants submit that a reasonable reading of *Nair* is limited to operations preformed among the “software protocol modules” of a transport, network, data link, and physical layer of a data protocol communications stack. In contrast, claim 1 is directed squarely to operations performed by a “higher level application” to process messages. The Examiner appears to agree with this point:

Although Applicant's cited passages of the reference are correct, they do not have any bearing as to how Nair teaches the claimed limitations.

Final Office Action, p. 8. However, the Examiner goes on to suggest that

Applicants must see that “as an initial step, a driver or physical layer protocol software module [read server software application] receives a frame of data” (p. 3, ¶ 23). Then “the driver processes the data frame” (p. 3, ¶ 23). The allocation of the buffer occurs after the frame is received.

Final Office Action, p. 8. Frankly, arguing that the “driver or physical layer protocol software module” is the same as the “server software application” makes no sense. *Nair* expressly distinguishes the driver (corresponding to a physical layer protocol module) from the “higher level application.” Further, *Nair* expressly distinguishes the data link, network, and transport layers from the “higher level application.”

The distinction between the operations of the “software protocol modules” and those of the “higher level application” is readily apparent from *Nair*’s discussion of processing of inbound data frames. Specifically, *Nair* teaches that a buffer used by the “software protocol modules” of the TCP/IP protocol stack may be discarded (or returned to the buffer pool) once a frame is provided to a server application. On this point, *Nair* provides:

“[P]rocessing of the data frame continues up the protocol stack until processing of the data frame by the machine is completed. **At such time, the data is read from the buffer at 230 and, for example, provided to an application software program.** At this point, for example, **the buffer is no longer needed** for temporarily storing the data packets while the various protocol software modules in the protocol stack process the data frame.”

Nair, ¶ 28. Clearly, the operations performed by the server application are distinct from those used to manage a buffer within different layers of the protocol stack. The present claims, however, are directed to actions of a “server application” in processing data has been received over a socket connection, i.e., after the data is, in the words of *Nair*, “provided to an application software program.”

Thus, Applicants submit that *Nair* fails to disclose a method performed by a server application in processing messages that includes receiving, at a socket configured for a server application executing on a computer, data from a remote source via a network connection prior to allocating a buffer to contain the data.

Further, Applicants submit that *Nair*, *Beighe*, and *Putcha* do not teach or suggest the recited step of “determining a mode to obtain the buffer according to a buffer mode parameter supplied with a receive operation call.” The Examiner concedes that *Nair* and *Beighe* alone do not disclose this limitation, but suggests that the following passage from *Putcha* does:

In one aspect, a network communication device for directing data units over a communication network includes at least one input and/or output port arranged to receive and transmit data units, a plurality of buffer units divided into several sub-pools, and a buffer allocator for allocating buffer units between the sub-pools. The buffer allocator is arranged to determine a priority value for each sub-pool based on quality of service parameter for

each connection established at at least one input port. The buffer allocator is also arranged to determine a utilization value of the input port, and arranged to allocate buffer units for each sub-pool based on the priority value and based on the utilization value, wherein a minimal number of connections established at a most utilized port will suffer loss of data units while receiving the data units.

Putcha, 4:18-33. *Putcha* discloses “a network communication device” that includes a “buffer allocator for allocating buffer units between the sub-pools.” *Putcha*, 4:18-24. The buffer allocator “is arranged to determine a priority value” and “to determine a utilization value.” Thus, the buffer allocator is allocating the available buffer space based on priority and utilization values. Applicants respectfully submit that allocating buffers based on priority and utilization does not disclose determining a mode to obtain the buffer based on a buffer mode parameter supplied with a receive operation call. The buffer allocator of *Putcha* simply does not specify a mode of buffer acquisition; instead, buffers are allocated in a single/fixed manner that does not depend on the pool the buffer is acquired from.

Thus, although the Examiner asserts that:

Applicant has not sufficiently defined what is meant by a ‘buffer mode parameter’ in the claim and is therefore open to interpretation. By this rationale, *Putcha* does, in fact, disclose a "buffer mode parameter" which could be construed as the priority value for the sub pool of *Putcha*. By this rationale, the rejection is maintained.

Final Office Action, p. 8. However, claim 1 expressly recites that the “buffer mode parameter” is “supplied with a receive operation call” issued by the server application and further, that the “buffer mode parameter” “indicates a buffer acquisition method for acquiring a buffer to contain the data received from a remote source via the network connection.” The Examiner’s “rationale” that “*Putcha* does, in fact, disclose a ‘buffer mode parameter’ which could be construed as the priority value for the sub pool of *Putcha*” ignores the express limitations recited by claim 1.

This point is even stronger when considering claims 15 and 26, which ultimately depend from claims 9 and 20, respectively. These dependent claims further specify that the buffer is allocated from “storage owned by the sockets server application” or from

“system-supplied storage not owned by the sockets server application.” In both cases, the claim further characterizes the actions of the “sockets server application” and not those of a “software protocol modules.” Nevertheless, the Examiner suggests:

that “*Beighe* further discloses that buffer [sic] is allocated from storage owned by the sockets server application based on a value of the buffer mode parameter (i.e. direction) (col. 3, lines 10-50),”

See *Final Office Action*, p. 4. First of all, in regards to claims 1, 9, and 20, the Examiner concedes that “*Nair* in view of *Beighe* do not specifically disclose determine a buffer acquisition mode,” *Final Office Action*, p. 4. Respectfully, it is completely contradictory, therefore, for the Examiner to assert that the *Beighe* discloses that a buffer is allocated from storage owned by the sockets server application based on a value of the buffer mode parameter.

Moreover, the inherent “direction” in which data communication occurs is not the same as a “buffer mode parameter” that specifies to obtain a buffer from “storage owned by the sockets server application” or from “system-supplied storage not owned by the sockets server application.” The cited passages from *Beighe* 3:10-50, describe the operations of a cable modem storing data packets “in a buffer in memory system 28.” *Beighe*, 3:9-11. Even when combined with *Nair*, it makes no sense to suggest that the “higher level application” form *Nair* or the “sockets server application” recited by the present claims would have access to the “memory system 28” of a cable modem. Finally, there is no teaching or suggestion in the cited material that a buffer may be allocated from different memory storage, i.e., from “storage owned by the sockets server application” or from “system-supplied storage not owned by the sockets server application,” and instead, the only memory storage is “memory system 28,” without the distinctions recited by claims 15 and 26.

For all the foregoing reasons, Applicants submit that claims 1-7, 9-11, 14-15, 17-18, 20-22, 25, 26, and 28-29 are patentable over *Nair*, in view of *Beighe*, and in further

view of *Puchta*. Accordingly, Applicants respectfully request that this rejection be withdrawn.

Claims 12 and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Nair* in view of *Beighe* in view of Glasser et al. (U.S. No. 5,764,890) (hereinafter Glasser).

First, on its face the rejection is defective. As pointed out in Applicants' Response of July, 6, 2006, claim 12 depends from claim 9, and claim 23 depends from claim 20, thus these claims include all of the limitations of claim 9 and 20, respectively. The Examiner concedes that *Nair* in view of *Beighe* does not disclose all the limitations recited by independent claim 9 and 20. Thus, when the Examiner states “[r]eferring to claim 12, *Nair* in view of *Beighe* discloses the invention substantively as described in claim 9”, presumably, the Examiner still believes that “*Nair* in view of *Beighe* do not [sic] specifically disclose determining a buffer acquisition mode according to a buffer mode parameter with a receive operation call.” *Office Action* dated April, 6, 2006, p. 4, *Final Office Action* dated August 9, 2006, p. 4. Nevertheless, Applicants believe that the above discussion regarding claims 9 and 20 demonstrate that these claims are patentable over *Nair* in view of *Beighe* and *Puchta*. Thus, Applicants believe a detailed discussion of Glasser cited in regards to dependent claims 12 and 23 is unnecessary. Accordingly, Applicants respectfully request that this rejection be withdrawn.

Claim 19 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Nair* in view of *Beighe* in view of *Fry* (USPN 4,467,411). Like the rejection of claims 12 and 23, the rejection is defective on its face. As pointed out in Applicant's Response of July, 6, 2006, Claim 19 depends from claim 9, and thus includes all of the limitations recited by claim 9. The Examiner concedes that *Nair* in view of *Beighe* does not disclose all the limitations of independent claim 9. Thus, when the Examiner asserts “[r]eferring to claim 19, *Nair* in view of *Beighe* discloses the invention substantively as described in claim 9”, presumably, the Examiner still believes that “*Nair* in view of *Beighe* do not [sic] specifically disclose determining a buffer acquisition mode according to a buffer mode

parameter with a receive operation call.” *Office Action* dated April, 6, 2006, p. 4, *Final Office Action* dated August 9, 2006, p. 4. Nevertheless, Applicants believe that the above discussion regarding claims 1, 9, and 20 demonstrate that these claims are patentable over *Nair* in view of *Beighe* and *Puchta*. Thus, Applicants believe a detailed discussion of the *Fry* reference cited in regards to dependent claim 19 is unnecessary. Accordingly, Applicants respectfully request that this rejection be withdrawn.

Therefore, Applicants respectfully submit that *Nair*, *Beighe*, and *Putcha* do not disclose the recited element of “receiving, at a socket configured for a server application executing on a computer, data from a remote source via a network connection prior to allocating a buffer to contain the data.” Accordingly, Applicants submit that claims 1, 9, and 20, as well as the respective dependent claims, are allowable, and respectfully requests withdrawal of this rejection.

Conclusion

Having addressed all issues set out in the office action, Applicants respectfully submit that the claims are in condition for allowance and respectfully request that the claims be allowed.

If the Examiner believes any issues remain that prevent this application from going to issue, the Examiner is strongly encouraged to contact Gero McClellan, attorney of record, at (336) 643-3065, to discuss strategies for moving prosecution forward toward allowance.

Respectfully submitted, and
S-signed pursuant to 37 CFR 1.4,

/Gero G. McClellan, Reg. No. 44,227/

Gero G. McClellan
Registration No. 44,227
PATTERSON & SHERIDAN, L.L.P.
3040 Post Oak Blvd. Suite 1500
Houston, TX 77056
Telephone: (713) 623-4844
Facsimile: (713) 623-4846
Attorney for Applicants